Changes

- Added information about other models (Naive Bayes, a regular neural net, a convolutional net trained on a different data set, KNN, SVM)
- Added information about computer vision techniques used
- Added questions slide
- Added information about another computer vision technique tested

Team Team Name

CSCI 470/575 Machine Learning
Graphology model

Zachary Aparicio
Stephen Hamrick
Daniel Hanuszczak
Pooja Gangrade
Trevor Kerr

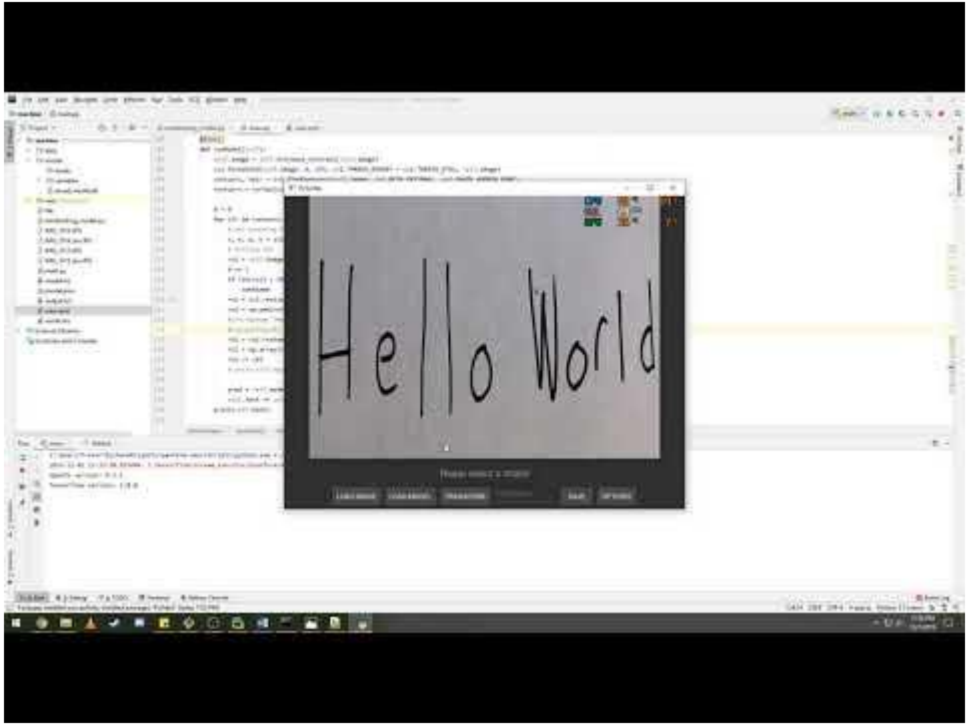
Problem

- Do you have notes that you want to digitize, but don’t want to type them by hand?
- Do you want to share notes with your friends but they can’t read your handwriting?
- Would you like to preserve your work for future reference?
- Currently, you have to transcribe by hand

Solution

- ML allows us to this much more quickly and accurately than other methods
- Product Workflow
    - Student/Teacher snaps picture of written text
    - Picture is input into our application through the GUI
    - Picture is divided into words/letters using Computer Vision
    - Machine Learning Model is applied and a text document is produced of the transcribed text

Demo



Assumptions, Constraints and Implications

Can only read texts written by sharpie pen on blank paper

Can not read symbols or special characters

Cursive writing is not supported

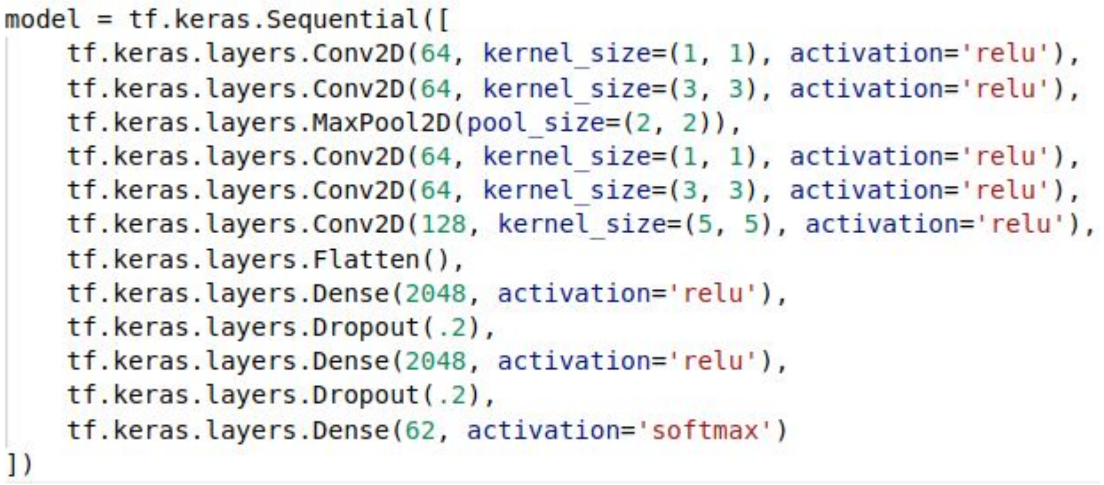
Spaces between characters will not be detected

How Solution was Built - Data

- EMNIST - 62 Classes
  - A-Z (26 classes)
  - a-z (26 classes)
  - 0-9 (10 classes)
- 600,000 Images

How Solution was Built - Best Model

- Convolutional Neural Net
    - 5 Convolutional Layers
    - Max Pooling - 2x2 pool size
    - 2 Dense Layers - 2048 Neurons, 20% Dropout
    - Output Layer - 62 Neurons, Softmax Activation
- 85% Accurate



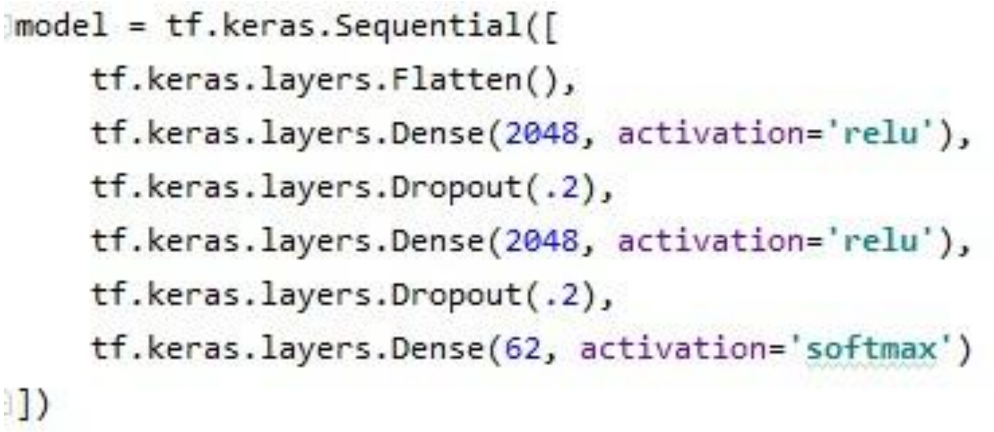
```
model = tf.keras.Sequential([
    tf.keras.layers.Conv2D(64, kernel_size=(1, 1), activation='relu'),
    tf.keras.layers.Conv2D(64, kernel_size=(3, 3), activation='relu'),
    tf.keras.layers.MaxPool2D(pool_size=(2, 2)),
    tf.keras.layers.Conv2D(64, kernel_size=(1, 1), activation='relu'),
    tf.keras.layers.Conv2D(64, kernel_size=(3, 3), activation='relu'),
    tf.keras.layers.Conv2D(128, kernel_size=(5, 5), activation='relu'),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(2048, activation='relu'),
    tf.keras.layers.Dropout(.2),
    tf.keras.layers.Dense(2048, activation='relu'),
    tf.keras.layers.Dropout(.2),
    tf.keras.layers.Dense(62, activation='softmax')
])
```

How Solution was Built - Computer Vision

- Need to extract text from scanned documents
- We used OpenCV
    - Threshold the image
    - Apply a gaussian blur
    - Dilate the image
    - Find contours
    - Extract each contour and create a bounding box
    - Make sure the contour is big enough and resize it
    - Use the model to predict the letter

How Solution Was Built - Other Solutions

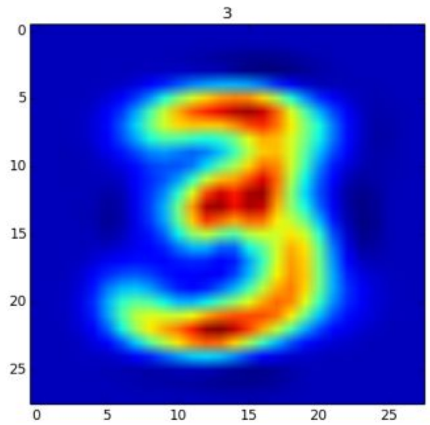
- Simple Neural Network
    - Only two Dense layers
- 80% accurate



```
model = tf.keras.Sequential([
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(2048, activation='relu'),
    tf.keras.layers.Dropout(.2),
    tf.keras.layers.Dense(2048, activation='relu'),
    tf.keras.layers.Dropout(.2),
    tf.keras.layers.Dense(62, activation='softmax')
])
```

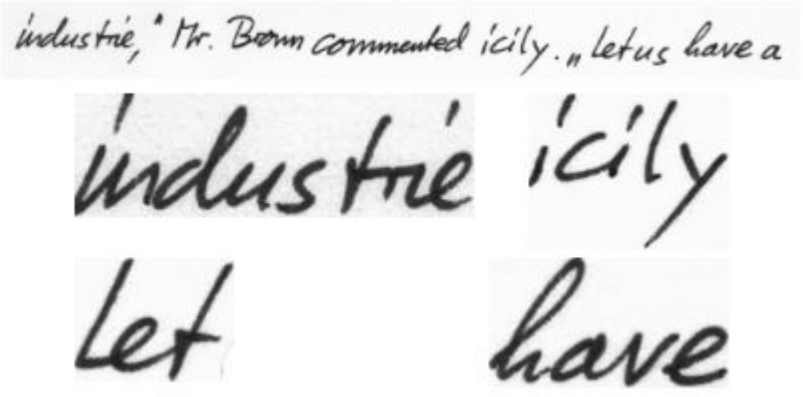
How Solution Was Built - Other Solutions

- Naive Bayes and Supervised Learning - 33% Accuracy
  - Search through the EMNIST Dataset
  - Find mean of all 62 classes with half of the data
  - Test on the second half of the data



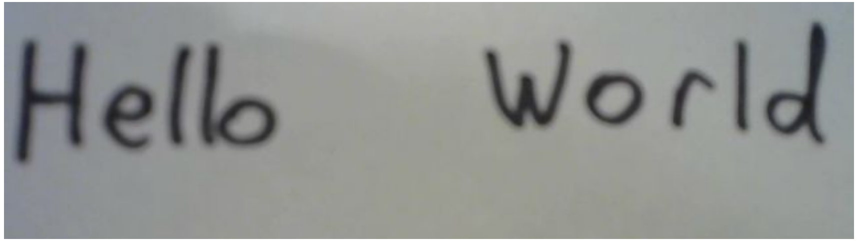
How Solution Was Built - Other Solutions

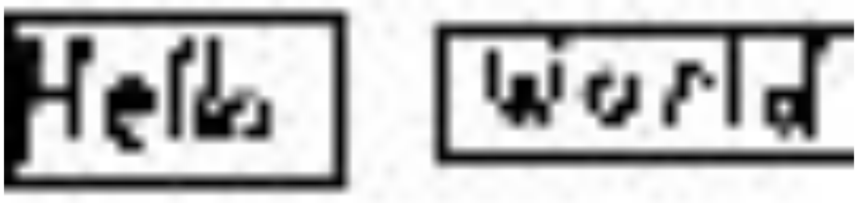
- IAM Handwritten document dataset
  - Requires document to be separated into words
  - Lower accuracy than using character detection
  - Utilized trained model found in https://github.com/githubharald/SimpleHTR



Other Solutions - Word Segmentation

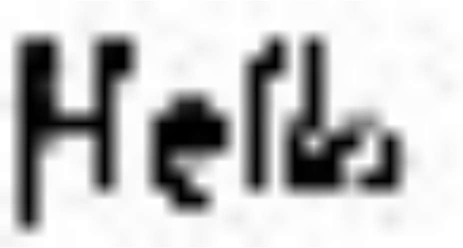
- Need for Word Segmentation
  - Current problem: No way to detect spaces between words because segmenting by character
  - Solution: Segment by word first, then segment each word into characters
  - Used word segmenting technique implemented in https://github.com/githubharald/WordSegmentation

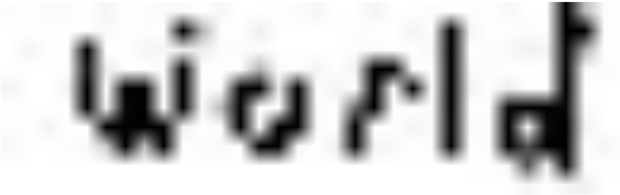




Other Solutions - Word Segmentation

- Problem with Word Segmentation
    - Segmented words using the selected method have very low resolution
    - Words are segmented from the left to the right of the image without taking into consideration the vertical axis on the page
        - Solution: documents will need to be separated into horizontal lines of text before segmenting into words





Other Solutions - KNN

- Features extracted using histogram of gradients
- Uses grid search to find optimal K value and weight function
  - Best K: 5
  - Weight function: distance
- About 73% accuracy

Other Solutions - SVM

- Similar to KNN method
- Features extracted using histogram of gradients
- Then transforms HOGs using PCA
- SVC is then used to train/predict
- About 52% accuracy

Summary

- Problem: How does one digitize handwritten text?
  - If not transcribing, it is nearly impossible.
- Solution: Use Computer Vision and Machine Learning Models to analyze
  - CV: Split into Words and Letters to analyze a letter at a time
  - ML: Deep Learning via CNN
- Issues that arose: Constraints and ML Complications
  - Dark, Handwritten Writing
  - Overfitting, Parameter Ambiguity

Questions?